Claims

What is claimed is:

1. A method of determining an allowable order of changes in a distributed system, the method comprising the steps of:

determining existing relationship descriptions between components of the system;

transforming acquired relationships into ordered tasks that are linked by temporal ordering constraints; and

creating an order of changes taking into account task relationship constraints.

2. The method of Claim 1, wherein the order of changes is sequential.

3. The method of Claim 1, wherein the order of changes is concurrent.

4. The method of Claim 1, further comprising refining an incoming request for change by breaking the request down into sub-requests.

5. The method of Claim 4, further comprising computing an allowable order of changes by interacting with the system.

6. The method of Claim 1, wherein creating the order of changes includes determining whether the ordered changes are conflicting and flagging such conflicts.

7. The method of Claim 1, wherein the changes are partially ordered.

8. The method of claim 1, wherein the changes are totally ordered.

9. The method of Claim 1, wherein the order of changes includes an estimate of the time required to complete a change.

10. The method of Claim 4, wherein the total change time is minimized by exploiting parallelism between change tasks.

11. The method of Claim 1, wherein the creation of the order of changes further takes into account a requested change management operation.

12. The method of Claim 1, wherein a requester identifies one or more target systems within the distributed system by name.

13. The method of Claim 12, wherein the names of the target systems are unique physical identifiers.

14. The method of Claim 12, wherein the names of the target systems are logical names which refer to one or more physical systems.

15. The method of Claim 1, wherein a requester does not identify one or more target systems within the distributed system by name.

16. The method of Claim 1, further comprising the steps of accessing and evaluating policy rules representing best practices.

17. The method of Claim 16, wherein the best practices include updating all affected software artifacts when a software artifact is updated.

18. The method of Claim 16, wherein the best practices include having a given set of software components installed on different systems.

19. The method of Claim 1, wherein one or more of the order of changes are persistently stored after being created.

20. The method of Claim 1, wherein a component is one of a service, an application, middleware, hardware, an operating system, a storage system, a network device, and a system associated with the computing environment.

21. A system for determining an allowable order of changes in a distributed system, the system comprising:

an arrangement for determining existing relationship descriptions between components of the system;

an arrangement for transforming acquired relationships into ordered tasks that are linked by temporal ordering constraints; and

an arrangement for creating an order of changes taking into account task relationship constraints.

22. The system of Claim 21, wherein the order of changes is sequential.

23. The system of Claim 21, wherein the order of changes is concurrent.

24. The system of Claim 21, further comprising an arrangement for refining an incoming request for change by breaking the request down into sub-requests.

25. The system of Claim 24, further comprising an arrangement for computing an allowable order of changes by interacting with the system.

26. The system of Claim 21, wherein creating the order of changes includes determining whether the ordered changes are conflicting and flagging such conflicts.

27. The system of Claim 21, wherein the changes are partially ordered.

28. The system of claim 21, wherein the changes are totally ordered.

29. The system of Claim 21, wherein the order of changes includes an estimate of the time required to complete a change.

30. The system of Claim 24, wherein the total change time is minimized by exploiting parallelism between change tasks.

31. The system of Claim 21, wherein the creation of the order of changes further takes into account a requested change management operation.

32. The system of Claim 21, wherein a requester identifies one or more target systems within the distributed system by name.

33. The system of Claim 32, wherein the names of the target systems are unique physical identifiers.

34. The system of Claim 32, wherein the names of the target systems are logical names which refer to one or more physical systems.

35. The system of Claim 21, wherein a requester does not identify one or more target systems within the distributed system by name.

36. The system of Claim 21, further comprising an arrangement for accessing and evaluating policy rules representing best practices.

37. The system of Claim 36, wherein the best practices include updating all affected software artifacts when a software artifact is updated.

38. The system of Claim 36, wherein the best practices include having a given set of software components installed on different systems.

39. The system of Claim 21, wherein one or more of the order of changes are persistently stored after being created.

40. The system of Claim 21, wherein a component is one of a service, an application, middleware, hardware, an operating system, a storage system, a network device, and a system associated with the computing environment.

41. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for determining an allowable order of changes in a distributed system, said method comprising the steps of:

determining existing relationship descriptions between components of the system;

transforming acquired relationships into ordered tasks that are linked by temporal ordering constraints; and

creating an order of changes taking into account task relationship constraints.